

SYSTEM, METHOD AND ARTICLE OF MANUFACTURE FOR UTILIZING EXTERNAL MODELS FOR ENTERPRISE WIDE CONTROL

COPYRIGHT NOTIFICATION

Portions of this patent application contain materials that are subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document, or the patent disclosure, as it appears in the Patent and Trademark Office.

FIELD OF THE INVENTION

This invention generally relates to improvements in computer systems, and more particularly, to system software for managing the design, simulation, implementation and maintenance of a manufacturing enterprise.

BACKGROUND OF THE INVENTION

This invention relates to electronic programmable controllers for operating industrial equipment and visualizing the industrial environment being controlled. Electronic programmable controllers utilize a programming language to develop control programs to control industrial equipment.

Programmable controllers are well-known systems for operating industrial equipment, such as assembly lines and machine tools, in accordance with a stored program. In these controllers, a stored program is executed to examine the condition of specific sensing devices on the controlled equipment, and to energize or de-energize selected operating devices on that equipment contingent upon the status of one or more of the examined sensing devices. The program not only manipulates single-bit input and output data representing the state of the sensing and operating devices, but also performs arithmetic operations, timing and counting functions, and more complex processing operations.

One industry that extensively uses programmable controllers is the automotive industry. In the automotive industry, various automotive parts are conveyed along machine lines consisting of many consecutive workstations. Most workstations include at least one tool that performs some function to alter the characteristics of workpieces as they are delivered to the station. For example, an unfinished cast engine block that requires a plurality of holes, bores, and threads, as well as other metal-removing procedures, may be provided at the beginning of a machine line that produces finished engine blocks. The machine line may consist of any number of different stations, each station performing a different procedure on the unfinished block. An indexer in the form of a transfer bar can be arranged to move each block from one station to the next following a completed process. Typically, at each station the block would be clamped prior to any metal-removing operation.

In this type of system, a programmable controller would receive inputs from all of the various tools at all of the workstations and would provide activating output signals to synchronize machine operation. During metal-removing periods with the transfer bar out of the way, all of the tools would perform their functions. In between metal-removing periods during transfer periods, the tools would be parked, the clamps unclamped, and the transfer bar would advance workpieces from one station to the next.

Industrial controllers are frequently programmed in Ladder Logic (LL) where instructions are represented graphically by "contacts" and "coils" of virtual relays connected

and arranged in ladder-like rungs across power rails. LL, with its input contacts and output coils, reflects the emphasis in industrial control on the processing of large amounts of input and output data.

LL also reflects the fact that most industrial control is "real time"; that is, an ideal industrial controller behaves as if it were actually composed of multiple relays connected in parallel rungs to provide outputs in essentially instantaneous response to changing inputs. Present industrial controllers do not, in fact, employ separate parallel relay-like structures, but instead simulate the parallel operation of the relays by means of a conventional Harvard or Von Neumann-type computer processor which executes instructions one at a time, sequentially. The practical appearance of parallel operation is obtained by employing extremely fast processors in the execution of the sequential control program.

As each rung is executed, inputs represented by the contacts are read from memory (as obtained from inputs from the controlled process or the previous evaluation of coils of other rungs). These inputs are evaluated according to the logic reflected in the connection of the contacts into one or more branches within the rungs. Contacts in series across a rung represent boolean AND logic whereas contacts in different branches and thus in parallel across the rung represent boolean OR logic.

Typically a single output coil at the end of each rung is set or reset. Based on the evaluation of that rung, this setting or resetting is reflected in the writing to memory of a bit (which ultimately becomes an output to the industrial process or to another LL rung).

Once a given rung is evaluated the next rung is evaluated and so forth. In the simplest form of LL programming there are no jumps, i.e. all rungs are evaluated in a cycle or "scan" through the rungs. This is in contrast to conventional computer programming where branch and jump instructions cause later instructions or groups of instructions to be skipped, depending on the outcome of a test associated with those branch or jump instructions.

While LL is well suited for controlling industrial processes like those in the automotive industry, LL programming is not an intuitive process and, therefore, requires highly skilled programmers. Where hundreds of machine tool movements must be precisely synchronized to provide a machining process, programming in LL is extremely time-consuming. The time and relative skill associated with LL programming together account for an appreciable percentage of overall costs associated with a control system. In addition, the final step in LL programming is typically a lengthy debugging and reworking step that further adds to overall system costs.

One way to streamline any type of programming is to provide predefined language modules, expressed in a language such as LL, which can be used repetitively each time a specific function is required. Because of the similar types of tools and movements associated with different machine-line stations, industrial control would appear to be an ideal industry for such language modules.

The predefined logic module approach works quite well for certain applications, like small parts-material handling or simple machining. The reason for this is that the LL required for these applications tends to be very simple. In small parts material handling applications the I/O count is low and the interfaces between modules are minimal. In fact, the mechanisms are often independent units, decoupled from neighboring mechanisms by part buffers such that no signals are required to be exchanged between modules. These "loosely